

9. Justin

Program Name: Justin.java

Input File: justin.dat

Justin's parents are considering buying a new car and when that happens he will get the current family car. He will finally have a car of his own! He wants to help his parents by creating a report that summarizes the purchase details for each of the cars they are considering.

They will be using a car loan to pay for most of the purchase so the information needed to help them make a decision is the monthly payment, the final cost of the car after making the loan payments and the amount of interest that will have been paid. Justin has found the following formula for calculating the monthly payment.

Formula	Legend
$p = \frac{ar}{1 - (1 + r)^{-n}}$	a = amount of loan r = monthly interest rate n = number of payments p = monthly payment

The loan amount a is the purchase price less the down payment. The monthly interest rate r is 1/12 of the APR or annual percentage rate for the loan as a decimal value not a percentage. The number of monthly payments n and monthly payment amount p are as expected. The monthly payment must be rounded to nearest penny; otherwise, partial pennies add up to extra pennies after all payments.

For each car, calculate and display the monthly payment, final cost which is the total of all payments plus the down payment, and the amount of interest paid which is the total of payments less the purchase price.

Using first set of data as a sample:

Monthly payment is \$587.07
 Final cost is \$37224.20 [587.07 * 60 + 2000.00]
 Interest paid is \$6236.79 [37224.20 – 30987.41]

Input: The first line contains an integer T which is the number of test cases with $3 \leq T \leq 20$. Each of T following lines are one complete test case and will contain four pieces of data:

Purchase price between and including \$10,000.00 and \$100,000.00
 Down payment d with $0.00 \leq d \leq 5000.00$
 APR as a percentage with $1.0000\% \leq APR \leq 20.0000\%$
 Number of monthly payments n with $12 \leq n \leq 96$

Output: One line for each test case containing 3 values: monthly payment, final cost, and total interest paid. Each value is rounded to 2 decimal places and has a leading \$ with the value right-aligned in a field of 9 columns. One space separates the fields.

Sample input:

```
3
30987.41 2000 7.95 60
65432.78 3000 9.375 84
15937.07 1000 15.66 48
```

Sample output:

```
$ 587.07 $ 37224.20 $ 6236.79
$ 1016.41 $ 88378.44 $ 22945.66
$ 420.72 $ 21194.56 $ 5257.49
```